A SYNTHETIC VIDEO DATASET FOR VIDEO COMPRESSION EVALUATION

Di Ma, Angeliki V. Katsenou, David R. Bull

Department of Electrical and Electronic Engineering, University of Bristol, BS8 1UB, UK
{di.ma, angeliki.katsenou, dave.bull}@bristol.ac.uk

ABSTRACT

In this paper, a new Synthetic video Texture dataset (SynTex) is introduced. It was generated using a Computer Graphics Imagery (CGI) environment and offers the capability of being able to generate many versions of the same scenes with different video parameters. This will support research in video compression enabling researchers to understand and model the relationship between video content and its coding parameters. To validate that SynTex is suitable for this purpose, firstly, typical spatio-temporal descriptors were calculated and compared against existing real video datasets with similar parameters. Then, the encoding statistics of SynTex were extracted using the HEVC reference software and compared to natural video datasets. The comparisons show that SynTex exhibits a comparable coverage over the spatial and temporal domain and it has similar encoding statistics to real video datasets.

Index Terms— Synthetic Video Dataset, Video Texture, Video Compression, HEVC Coding Statistics.

1. INTRODUCTION

With increasing demands for more immersive video formats, the amount of video data that needs to be compressed while preserving its quality is massive [1]. Cisco reports that by 2021 82% of the consumer traffic will be video data [2]. What is more important is that the trade-off between compression efficiency and quality is content related. This poses the challenge of better comprehending content properties and their relation to compression. Textured areas, and particularly dynamic textures, can be challenging to compress. Furthermore, different types of textures have different coding performance [3, 4]. In order to fully understand explore and model the relation of texture to compression, we need a dataset that contains the same content but with different acquisition (e.g. camera motion) or content parameters (e.g. different level of coarseness).

There are few freely available video datasets that contain a variety of video textures [4–6]. One of the most cited is DynTex [5]. This contains 650 PAL resolution dynamic video sequences (spatial resolution 720×576 at 25 frames per second (fps)) with a wide range of texture types. It has been widely used for developing dynamic texture classification and recognition algorithms. However, the video parameters (spatial resolution and frame rate) are obsolete compared to current requirements. Thus, the use of DynTex in future research is limited. HomTex [4, 7] and BVI-Texture [6] datasets were developed with the aim of analysing the properties of video texture and its coding performance. The BVI-Texture dataset [6] contains 20 video sequences with Full High Definition (FHD) resolution (1920×1080) at a frame rate of 60 fps. This dataset was used to test HEVC compression efficiency and its perceptual quality versus bit rate performance. It has been also used to develop video quality assessment metrics and future video coding algorithms. Although BVI-Texture dataset satisfies current video specifications, the small number of available sequences is not adequate for an extensive analysis. Another available dataset containing video textures is HomTex [4]. HomTex contains 120 video sequences that are manually cropped and selected from the DynTex and BVI-Texture datasets to obtain homogeneous video textures. The resolution and frame rate are 256×256, 25 fps and 60 fps, respectively.

In all of the above mentioned datasets, the number of sequences is insufficient to allow full exploration and understanding of the video parameter space. A larger video dataset that contains many different variations (different camera motions, frame rates, spatial resolutions, etc.) of static and dynamic textures is thus needed. Due to the enormous number of potential combinations of parameters related to video content (e.g. spatial patterns, colourfulness, complex local motion patterns) and acquisition (e.g. frame rate, shutter angle, camera position), the cost in personhours of capturing multiple variants is prohibitive. Also the randomness of some textural content (e.g. foliage, falling leaves), makes capturing an identical scene with different video parameters infeasible. Drawing inspiration from computer vision [8–11], we propose to address this by the generation of a synthetic video dataset. Such an approach has the benefit of using parameterised models for the production of the synthetic video content. This translates to datasets that are reproducible and can densely cover the video parameter space.

The use of synthetic data is a common practise in many research areas, especially in situations where real data may be difficult to acquire, due to budget, time or privacy concerns. For example, in computer vision, synthetic data are used for scene understanding [9] or object recognition [8] and have been proven reliable and useful especially for training neural networks [10, 11]. To the best of our knowledge, in the field of video compression, there is only one synthetically-generated dataset [12], designed to simulate a multi-lens stereoscopic video system with the aim to be used for multi-view compression, streaming, or other computer graphics related research.

This paper introduces a Synthetic video Texture dataset (SynTex) with the aim of studying and analysing video coding performance for different video textures and parameters. To the best of our knowledge, this is the first synthetic video texture dataset that has been developed for this purpose. It is developed using a Computer Graphics Imagery (CGI) environment and is based on the generation of 3D models to which are rendered and projected to capture artificial video content. After creating the SynTex dataset, we first calculate and compare the range of low-level features of the uncompressed video content with existing real video datasets that include a high textural content, BVI-Texture [6], BVI-HFR [13] and HomTex [4]. Then, we compress the sequences and compare the coding performance of SynTex and HomTex to further validate the effective-

The authors acknowledge EPSRC for funding (grant #: EP/L016656/1, EP/M000885/1) and the Leverhulme Trust.

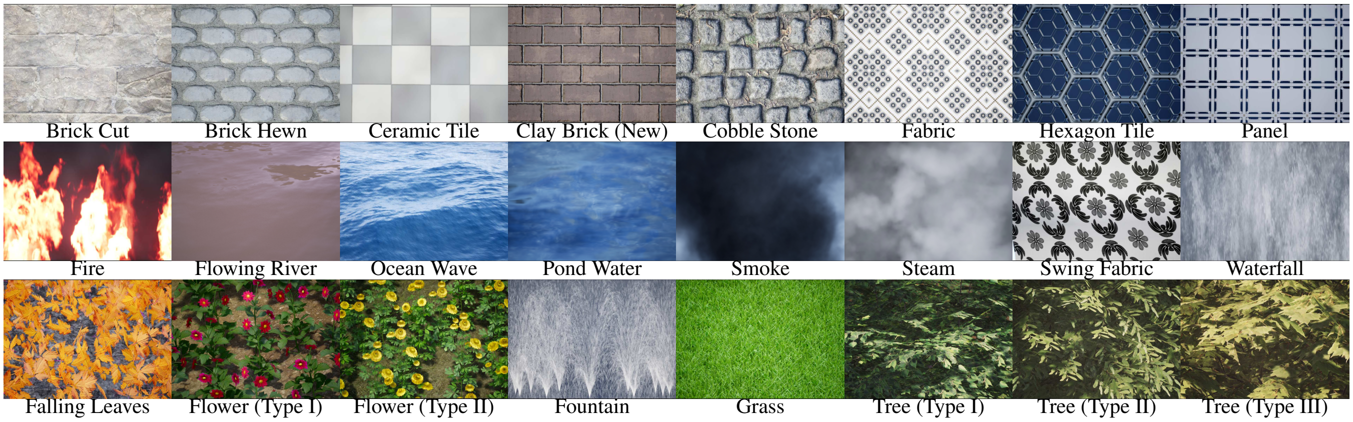

Fig. 1: Sample frames from the SynTex dataset (the top row is 'Static' textures, the middle row is 'Dynamic Continuous' textures, and the bottom row is 'Dynamic Discrete' textures).

Table 1: Description of scenes and parameters of SynTex dataset.

Video Texture	Type of Scene	Model Parameters
Static	Brick Cut; Brick Hewn; Ceramic Tile; Clay Brick (New); Cobble Stone (Rough); Fabric; Panel; Hexagon Tile	Granularity; Camera Motion
Dynamic Continuous	Fire; Flowing River; Ocean Wave; Pond Water; Smoke; Steam; Swing Fabric; Waterfall	Spread Width of Fire; Wave Speed; Intensity of Smoke; Initial Velocity of Smoke; Initial Velocity of Steam; Different Wind Direction of Fabric; Swing Speed; Density of Waterfall; Granularity
Dynamic Discrete	Falling Leaves; Flower; Grass; Tree; Fountain	Different types of Leaves/Flower/Tree; Wind Speed; Density of Fountain; Spread Width of Fountain; Granularity

ness of SynTex. The results show that SynTex covers a wide range of low-level features and has a similar coding performance when compared to real video datasets. It is thus of use for further understanding of compression performance and as a basis for designing new modes or training machine-learning based encoders.

The remainder of this paper is organised as follows: Section 2 describes the SynTex dataset and its parameters. Section 3 reports on the comparison of SynTex to other real video datasets and validates its effectiveness. Conclusions are finally drawn in Section 4.

2. DESCRIPTION OF THE SYNTEX DATASET

SynTex was created using the CGI tool, Unreal Engine 4 (UE4) [14]. UE4 is a C++ based tool that is widely used by the games industry and also by movie makers. It has, however, also recently been used for research purposes, for example for the analysis of stereo vision [9] and studying virtual reality (VR) [15]. UE4 has a variety of assets that include models for different scenes and objects. For each of these models there is a set of different parameters that can be adjusted, for example the amplitude or the speed of a wave. UE4 also includes universal parameters for capturing video that simulate a real camera, such as resolution, frame rate, shutter angle and viewing angle. These parameters will be used in our future work to create a synthetic video dataset that aims at simulating real scenes captured at different frame rates, with a different shutter angle, etc.

Our synthetic video dataset [16] contains 196 homogeneous video sequences [4] with spatial resolution of 1920×1080, frame rate at 60 fps, and 180° shutter angle. The selection of these parameters was driven by the typical requirements for video content and also by the need for common parameters with existing datasets (so as to be able to validate the generated video sequences). The added value of using this methodology is that this dataset can be easily captured using different parameters, e.g. different frame rate.

Based on previous studies on the analysis of video content for video compression purposes [4,17], video textures are classified into three types, static (e.g. a camera panning over a still scenery) and dynamic continuous (e.g. a scene of ocean waves) and dynamic discrete (e.g. a scene of moving foliage). In this paper, we followed the same definitions to generate synthetic video textures. Sample frames of the generated video sequences are illustrated in Fig. 1 and the differentiating parameters per different model are reported in Table 1. As mentioned above, the video acquisition parameters are the same for all videos. Also, some general parameters such as the texture granularity and amount of motion were uniform for the different versions of the videos (wherever it was applicable). Table 1 and Figure 1 show that there are eight scenes for each type of video texture. Each scene has a wide range of associated parameters that can be flexibly modified.

3. VALIDATION OF THE SYNTEX DATASET

As the video content in SynTex is artificial, in order to validate its effectiveness, we must confirm that SynTex has similar properties when compared to real video textures. First, we compare the content characteristics of the uncompressed videos and then we compress them and compare their compression performances and statistics. We compare SynTex against the published datasets BVI-HFR [13], BVI-Texture [6] and HomTex [4], that contain a significant quantity of real textures. We emphasise that we expect deviations in the examined statistics due to the following reasons. First, the real datasets have a smaller number of video sequences (20, 22, and 120, respectively), and BVI-HFR and HomTex (20/120 sequences) are captured at different frame rates. Moreover, BVI-HFR and BVI-Texture have heterogeneous textural content, which will result in different low-level features. Furthermore, although the datasets all provide good coverage, the actual content is very different (the spatio-temporal patterns cannot be identically replicated) and are expected to have different statistics.

3.1. Low-level Features

In this section, we compare the coverage of low-level features: Spatial Information (SI), the Motion Vectors (MV), the Colourfulness (CF) and the Temporal Information (TI) of SynTex with BVI-HFR,

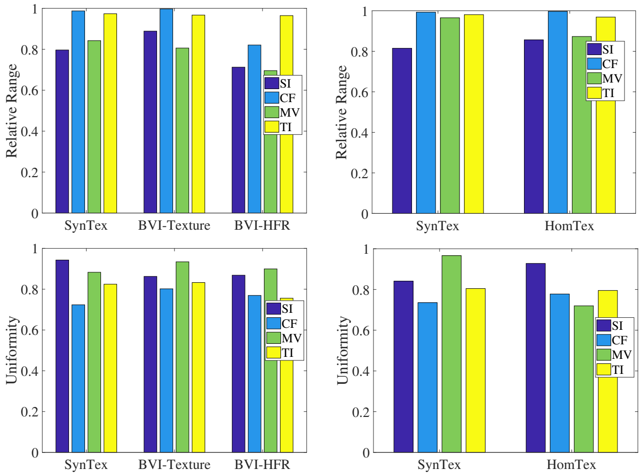

Fig. 2: The relative range and uniformity of coverage of SynTex and the compared datasets (left column: 1920×1080 spatial resolution datasets, right column: 256×256 spatial resolution datasets).

Table 2: The relative total coverage of SynTex and other datasets.

	SynTex	BVI-Texture	BVI-HFR	HomTex
Rel. Total Coverage	**0.40**	0.35	0.25	0.33

BVI-Texture, and HomTex datasets, as suggested by Winkler [18]. Since the spatial resolution of SynTex is 1920×1080 at a frame rate of 60 fps, it was directly compared with BVI-Texture and BVI-HFR datasets (60 fps version). However, in order to compare it with HomTex, each sequence in SynTex had to be downscaled to 256×256.

Figure 2 shows the relative ranges and uniformity of coverages of SynTex compared to BVI-Texture and BVI-HFR on the left and the relative ranges and uniformity of coverages of the downscaled SynTex and HomTex on the right, as defined in [18]. As can be seen, the relative ranges of CF, MV and TI are very close to the other datasets. The relative range value of SI in SynTex is slightly lower, which is expected due to the homogeneity of the video content. Regarding the uniformity of coverage values for SI, MV and TI of SynTex, they are comparable with the other datasets. It should be noted that the uniformity of MV for SynTex is significantly higher than in HomTex. This is explained by the higher temporal cohesion and homogeneity of the sequences. The motion patterns in SynTex are generated using models and this ensures a uniform motion across all frames. Last but not least, all above comparisons are confirmed by the relative total coverage of SynTex that is reported in Table 2. As highlighted in the table, SynTex achieves a better relative total coverage compared to BVI-Texture, BVI-HFR, and HomTex.

3.2. Video Coding Statistics

To further validate the effectiveness of SynTex, we use the HEVC reference codec (HM 16.2) [19] to compress SynTex at five quantisation levels, QP=$\{22, 25, 27, 32, 37\}$, and compare coding statistics with HomTex as in [4].

In Fig. 3, we illustrate examples of Peak Signal to Noise Ratio (PSNR) (dB) versus the required bits per pixel for similar video textures from SynTex and HomTex. Although the two datasets contain different texture patterns, we are showing examples of quality-rate curves of video textures similar both in spatial and temporal characteristics. Also, we note that in order to express the compression ratio, we use bits per pixel instead of bit rate, as this compensates for the different frame rate of HomTex and SynTex sequences. From the examples depicted in Fig. 3, we notice that for all three video texture types, the real and synthetic sequences have similar curves.

Table 3: Extracted coding statistics during the coding process [4].

Category	Statistics	Description
Prediction Modes	Intra(%), Skip(%), Merge(%), Inter(%)	Percentage of the predicted modes for Coding Unit (CU)
Partitioning	avgNumPart	Average of the number of partitions per Coding Transform Unit (CTU)
Bits	avgBits	Average of the number of bits per pixel
Distortion	avgDist	Average of the Sum of Absolute Differences (SAD) per pixel
Bit Allocation	bitsModeSignal (%) bitsPart (%) bitsIntraDir (%) bitsMergeIdx (%) bitsMotionPred (%) bitsResidual (%)	Percentage of bits used to encode mode selection, partitioning, intra modes, merge indexes, motion prediction, and residual signal
Residual Statistics	avgCorrResi	Average of the correlation between original and residual frames
Motion Vectors	avgLengthMV stdDistMV	Average length of motion vectors and standard deviation of motion vectors directions

Also, between the three texture types there is a clear difference in the compression efficiency. Static textures generally exhibit very high quality even at low bit rates, while dynamic discrete sequences require a much higher bit budget for the same quality.

Figure 4 shows the distribution of coding statistics of the downscaled SynTex and HomTex datasets. The results presented here are at a QP of 25 using the Random Access profile [20] as in [4]. 33 coding statistics were computed and are related to the prediction modes, the partitioning of the Coding Tree Units (CTUs), the residual information, the bits allocation, the motion vectors, the distortion, and the bit rate (see Table 3). Due to space limitations, in Figure 4 we are presenting 16 typical encoding statistics.

Prediction mode statistics: A first observation from Figure 4 is that, for both SynTex and HomTex, there are significant differences between the different types of video textures. Static textures exhibit a high percentage of Skip modes and a low percentage of Intra modes, mainly because they only have simple camera motion with a fixed movement direction [21]. Intra mode is also mainly used to encode dynamic textures. A noticeable difference between SynTex and HomTex is in the Intra and Skip modes. This is explained by the different acquisition parameters of SynTex and by the fact that many of sequences from HomTex suffer from noise, as also explained in [17]. Another reason that explains the differences in Skip statistics for the dynamic continuous textures is the frame rate. SynTex is captured at 60 fps which is higher than that of HomTex (the frame rate for most continuous textures in HomTex is only 25 fps).

Partitioning statistics: It can be seen from Fig. 4 that the distribution of SynTex is within the value range of HomTex, and for both of these two datasets, different types of video textures also have different distributions. Static textures have the lowest number of partitions (medians of 2 and 4 partitions per CTU for SynTex and HomTex respectively). The highest number of partitions for dynamic discrete (medians of 21 and 36 partitions per CTU for SynTex and HomTex respectively). The reason for this is that there are many regular objects or regions in discrete textures, hence the higher number of partitions per CTU will be obtained [22].

Bit statistics: The regularity of distributions of bits for SynTex and HomTex are consistent. In general, since static textures exhibit simplicity of motion and texture type, it will have the lowest average number of bits per pixel [4, 23]. Dynamic discrete textures exhibit more random local motion and variability in shapes, hence the average number of bits per pixel is higher compared to other texture types [4, 23]. Dynamic continuous textures have irregular regions and complex motions, hence the average number of bits per pixel will be also higher than static [4, 24].

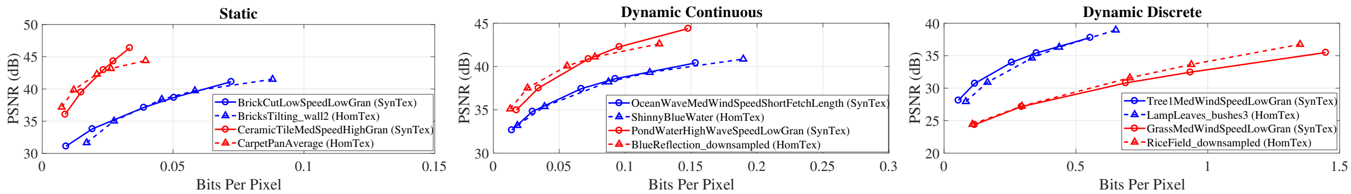

Fig. 3: Examples of PSNR-bits per pixel curves for sequences with similar textural content from SynTex and HomTex encoded by HEVC reference software. The solid lines depict sequences from the SynTex dataset and the dashed lines sequences from HomTex.

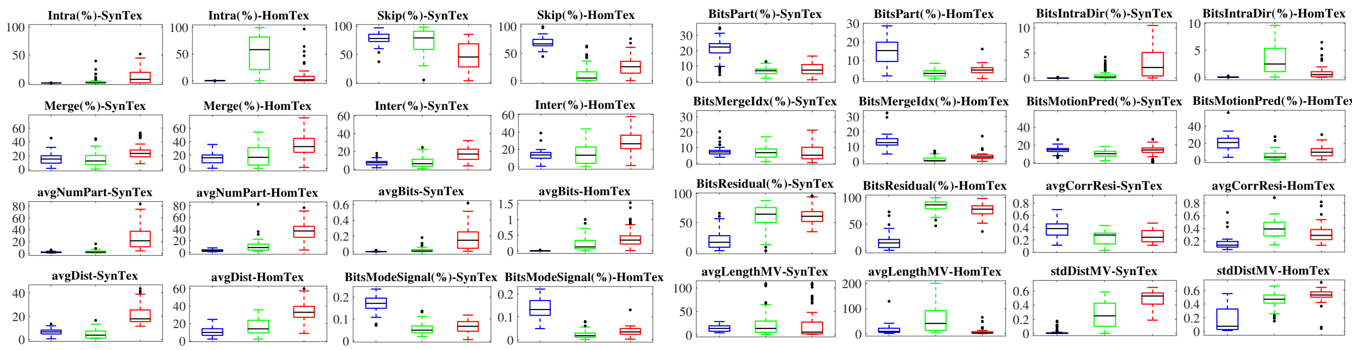

Fig. 4: Boxplots of coding statistics of the downscaled SynTex and HomTex (blue for static, green for dynamic continuous and red for dynamic discrete).

Distortion statistics: The distributions of the distortion statistic for SynTex is consistent with HomTex. Figure 4 shows that the averages of the SAD per pixel of dynamic continuous and dynamic discrete are higher than static textures because the motions of dynamic textures are highly complex and random compared to static textures, so that they have higher residual energy compared to static textures [4, 25]

Bits allocation statistics: Figure 4 shows that the distributions of bits allocation for SynTex and HomTex are consistent. In general, for SynTex and HomTex, most bits are allocated to residual encoding in dynamic continuous and discrete textures, this indicates that due to the complex textures and motions, a higher percentage of bits are needed to encode residuals [20]. For static textures, due to the simplicity of motions and textures, the bits used to code are evenly distributed and this also indicates a better prediction result in static textures compared to dynamic textures [4].

Residual statistics: Figure 4 shows that the residual statistics vary with texture types. Also, the distribution of SynTex is within the value range of HomTex. It should be noted that the average correlation of the residual statistic for static textures of SynTex is slightly higher than HomTex. This is because SynTex contains more speed variations than HomTex. The higher the camera motion, the higher residual energy will be obtained [26, 27]. Furthermore, SynTex contains versions of static textures with different granularity that are spatially complex, such as Brick Cut, Fabric, Cobble Stone (Rough), etc. On the other hand, HomTex only contains rather smooth texture types (e.g. sky, ceiling). Hence, SynTex has a higher residual energy than HomTex.

Motion Vector statistics: For both SynTex and HomTex, static textures have short motion vector length and smaller standard deviation of directions. This shows that static textures have higher motion consistency. It should be noted that, on the one hand, the standard deviation of motion directions of SynTex is within the value range of HomTex. On the other hand, for static textures, SynTex is much lower than the median value of HomTex. The reason is that since the static textures in SynTex are captured in a controlled environment, i.e. there is no camera shake during the capturing process and no slight change of movement direction for the camera, which commonly exists in real video shooting [1]. Hence, the direction of motion vectors of static textures in SynTex is strictly consistent with each other compared to HomTex. Dynamic continuous textures have the largest magnitude of motion vectors and discrete textures present a slightly smaller motion vector length with highest standard deviation of motion directions due to the higher percentage of random motions [4]. For both SynTex and HomTex, dynamic continuous and discrete textures achieve a higher standard of deviation in the motion vectors directions than static textures, due to the higher percentage of random motions.

To sum up, most distributions of the encoding statistics of SynTex are consistent with HomTex. The reasons for the few deviations are due to the different spatio-temporal patterns (diversity, regularity, granularity) and acquisition parameters (different camera speed, frame rate, shutter angle). However, these differences between SynTex and HomTex are small and we conclude that the rate distortion properties of SynTex in HEVC are sufficiently close to real video textures, to make a credible proxy.

4. CONCLUSION

In this paper, we introduced a synthetic video texture dataset, SynTex. SynTex covers a wide range of video textures, and spatial and temporal patterns, and allows the parameters related to video content (e.g. granularity, wind speed, camera speed) to be flexibly modified. The coverage of important low-level features and coding performance of SynTex are shown to be similar to real video datasets. Thus, we can conclude that SynTex has similar properties compared to real videos. SynTex is the first video texture dataset created for video compression purposes. It can be used by researchers to analyse and understand the video parameter space and its relation to video compression. One of the benefits of this synthetic dataset is that by using UE4, it can be further extended to include more video variations (different scene, heterogeneous content, etc.) to cover the needs of training, testing and validating of any video acquisition, analysis and compression research method. Part of the future work is to perform a subjective evaluation of SynTex at the different compression levels to study the relationship of perceptual quality and video content parameters.

5. REFERENCES

[1] D. R. Bull, *Communicating pictures: A course in Image and Video Coding*, Academic Press, 2014.

[2] White Paper, "Cisco Visual Networking Index: Forecast and Methodology, 2016 – 2021," Tech. Rep. 1465272001663118, CISCO, Sep. 2017.

[3] A. V. Katsenou, M. Afonso, D. Agrafiotis, and D. Bull, "Predicting video rate-distortion curves using textural features," in *Picture Coding Symposium (PCS)*. IEEE, 2016, pp. 1–5.

[4] M. Afonso, A. Katsenou, F. Zhang, D. Agrafiotis, and D. Bull, "Video texture analysis based on HEVC encoding statistics," in *Picture Coding Symposium (PCS)*. IEEE, 2016, pp. 1–5.

[5] R. Péteri, S. Fazekas, and M. J. Huiskes, "DynTex: A comprehensive database of dynamic textures," *Pattern Recognition Letters*, vol. 31, no. 12, pp. 1627–1632, 2010.

[6] M. A. Papadopoulos, F. Zhang, D. Agrafiotis, and D. Bull, "A video texture database for perceptual compression and quality assessment," in *IEEE International Conference on Image Processing (ICIP)*, Sept 2015, pp. 2781–2785.

[7] A. V. Katsenou, M. Afonso, and D. R. Bull, "VIL: Homogeneous Video Texture Dataset (HomTex)," https://data.bris.ac.uk/data/dataset/1h2kpxmxdhccf1gbi2pmvga6qp, Jul. 2016.

[8] A. Gaidon, Q. Wang, Y. Cabon, and E. Vig, "Virtual worlds as proxy for multi-object tracking analysis," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 4340–4349.

[9] Y. Zhang, W. Weichao Qiu, H. X. Chen, Q., and A. Yuille, "UnrealStereo: Controlling hazardous factors to analyze stereo vision," in *International Conference on 3D Vision (3DV)*. IEEE, 2018.

[10] A. Dai, A. X. Chang, M. Savva, M. Halber, T. A. Funkhouser, and M. Nießner, "ScanNet: Richly-annotated 3D reconstructions of indoor scenes.," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, vol. 2, pp. 5828–5839.

[11] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 3234–3243.

[12] F. Zhang, W.-C. Feng, and F. Liu, "A multi-lens stereoscopic synthetic video dataset," in *ACM Mutlimedia Systems*, 2015.

[13] A. Mackin, F. Zhang, and D. R. Bull, "A study of subjective video quality at various frame rates," in *IEEE International Conference on Image Processing (ICIP)*, Sept 2015, pp. 3407–3411.

[14] "The Unreal Engine (UE4)," https://www.unrealengine.com, 2018.

[15] U. C. Pendit, M. B. Mahzan, M. D. F. B. M. Basir, M. B. Mahadzir, and S. N. binti Musa, "Virtual reality escape room: The last breakout," in *Information Technology (INCIT), 2nd International Conference on*. IEEE, 2017, pp. 1–4.

[16] D. Ma, A. V. Katsenou, and D. R. Bull, "VIL: Synthetic Video Texture Dataset - SynTex," https://data.bris.ac.uk/data/dataset/24imj6d9s27me2n211cf6jxkio, Dec. 2018.

[17] A. V. Katsenou, T. Ntasios, M. Afonso, D. Agrafiotis, and D. R. Bull, "Understanding video texture - a basis for video compression," in *Multimedia Signal Processing (MMSP), IEEE 19th International Workshop on*. IEEE, 2017, pp. 1–6.

[18] S. Winkler, "Analysis of public image and video databases for quality assessment," *IEEE Journal of Selected Topics in Signal Processing*, vol. 6, no. 6, pp. 616–625, Oct 2012.

[19] J.-R. Ohm, G. J. Sullivan, H. Schwarz, T. K. Tan, and T. Wiegand, "Comparison of the coding efficiency of video coding standards-including high efficiency video coding (HEVC)," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1669–1684, 2012.

[20] G. J. Sullivan, J.-R. Ohm, W.-J. Han, T. Wiegand, et al., "Overview of the high efficiency video coding (HEVC) standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, 2012.

[21] J. Kim, J. Yang, K. Won, and B. Jeon, "Early determination of mode decision for HEVC," in *Picture Coding Symposium (PCS)*. IEEE, 2012, pp. 449–452.

[22] S. Cho and M. Kim, "Fast CU splitting and pruning for suboptimal CU partitioning in HEVC intra coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 9, pp. 1555–1564, 2013.

[23] H. Sun, S. Gao, and C. Zhang, "Adaptive bit allocation scheme for rate control in high efficiency video coding with initial quantization parameter determination," *Signal Processing: Image Communication*, vol. 29, no. 10, pp. 1029–1045, 2014.

[24] J. Si, S. Ma, and W. Gao, "Efficient bit allocation and ctu level rate control for high efficiency video coding," in *Picture Coding Symposium (PCS)*. IEEE, 2013, pp. 89–92.

[25] J. Vanne, M. Viitanen, T. D. Hamalainen, and A. Hallapuro, "Comparative rate-distortion-complexity analysis of HEVC and AVC video codecs," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1885–1898, 2012.

[26] G. Tian and S. Goto, "Content adaptive prediction unit size decision algorithm for HEVC intra coding," in *Picture Coding Symposium (PCS)*. IEEE, 2012, pp. 405–408.

[27] Y. Wang, X. Fan, D. Zhao, and W. Gao, "Mode dependent intra smoothing filter for HEVC," in *IEEE International Conference on Image Processing (ICIP)*. IEEE, 2016, pp. 539–543.